



Clean Code: A Handbook of Agile Software Craftsmanship

Robert C. Martin

[Download now](#)

[Read Online](#) ➔

Clean Code: A Handbook of Agile Software Craftsmanship

Robert C. Martin

Clean Code: A Handbook of Agile Software Craftsmanship Robert C. Martin

Even bad code can function. But if code isn't clean, it can bring a development organization to its knees. Every year, countless hours and significant resources are lost because of poorly written code. But it doesn't have to be that way.

Noted software expert Robert C. Martin presents a revolutionary paradigm with *Clean Code: A Handbook of Agile Software Craftsmanship*. Martin has teamed up with his colleagues from Object Mentor to distill their best agile practice of cleaning code “on the fly” into a book that will instill within you the values of a software craftsman and make you a better programmer—but only if you work at it.

What kind of work will you be doing? You'll be reading code—lots of code. And you will be challenged to think about what's right about that code, and what's wrong with it. More importantly, you will be challenged to reassess your professional values and your commitment to your craft.

Clean Code is divided into three parts. The first describes the principles, patterns, and practices of writing clean code. The second part consists of several case studies of increasing complexity. Each case study is an exercise in cleaning up code—of transforming a code base that has some problems into one that is sound and efficient. The third part is the payoff: a single chapter containing a list of heuristics and “smells” gathered while creating the case studies. The result is a knowledge base that describes the way we think when we write, read, and clean code.

Clean Code: A Handbook of Agile Software Craftsmanship Details

Clean Code: A Handbook of Agile Software Craftsmanship

466 pages , Published (first published January 1st 2007)



[Download Clean Code: A Handbook of Agile Software Craftsmanship ...pdf](#)



[Read Online Clean Code: A Handbook of Agile Software Craftsmanship ...pdf](#)

Download and Read Free Online Clean Code: A Handbook of Agile Software Craftsmanship Robert C. Martin

Oana Sipos says

These are rather notes than a review while reading:

1. Use very descriptive names. Be consistent with your names.
2. A function should not do more than one thing.
3. SRP (Single Responsibility Principle): a class or module should have one, and only one, reason to change.
4. Stepdown rule: every function should be followed by those at the next level of abstraction (low, intermediate, advanced).
5. A long descriptive name is better than a short enigmatic name. A long descriptive name is better than a long descriptive comment.
6. The ideal number of arguments for a function is zero (niladic). Next comes one (monadic), followed closely by two (dyadic). Three arguments (triadic) should be avoided where possible. More than three (polyadic) requires very special justification and then shouldn't be used anyway.
7. Flag arguments are ugly. Passing a boolean into a function is loudly proclaiming that this function does more than one thing. It does one thing if the flag is true and another one if the flag is false.
8. Write learning test when using third-party code to make sure it behaves the way you expect it to. And if codebase changes in time, at least you find out early enough.

Vladimir says

This book makes some very good points, sometimes taking them to extreme ("Never write functions longer than 15 lines! Never write functions with more than three arguments!"). Some of these points were quite new and useful for me - YMMV. It's too Java-specific in a few places, and reading the last refactoring chapter on a kindle was quite a challenge, but otherwise it was well worth a read. At least I got a clear picture of how I want to refactor a big piece of my current project after reading this :)

Francis Fish says

The first half of this book is well worth a read. Then I was reminded of Martin Fowler's (I think) comment that the original Design Patterns Elements of Reusable Software book was a response to the limitations of C++. It dovetailed so well into Java because Java has a lot of the same annoying limitations, and in some ways is even harder.

The latter section of the book contains some worked examples that I didn't always agree with because they seemed to be totally over done. A lot of the refactorings came from limitations in the language and even then felt arbitrary and not that "clean", more like differences of opinion.

In light of this I think the book would have been better titled *Clean Java*, and then we'd all know where we stand. Have to say I was disappointed by the case studies. I think if you're a jobbing Java programmer you will get a real benefit from this book. I use dynamic languages like Ruby and most of the problems described in need of refactoring just never happen.

From Reader Review Clean Code: A Handbook of Agile Software Craftsmanship for online ebook

From reader reviews:

Ana Lopez:

Typically the book Clean Code: A Handbook of Agile Software Craftsmanship will bring someone to the new experience of reading a new book. The author style to explain the idea is very unique. In the event you try to find new book to learn, this book very suited to you. The book Clean Code: A Handbook of Agile Software Craftsmanship is much recommended to you to see. You can also get the e-book from the official web site, so you can quicker to read the book.

Benjamin Ward:

As we know that book is important thing to add our knowledge for everything. By a reserve we can know everything we wish. A book is a list of written, printed, illustrated as well as blank sheet. Every year has been exactly added. This book Clean Code: A Handbook of Agile Software Craftsmanship was filled concerning science. Spend your extra time to add your knowledge about your research competence. Some people has several feel when they reading the book. If you know how big selling point of a book, you can experience enjoy to read a guide. In the modern era like at this point, many ways to get book that you just wanted.

Rose Duprey:

As a college student exactly feel bored to be able to reading. If their teacher inquired them to go to the library in order to make summary for some publication, they are complained. Just tiny students that has reading's heart or real their interest. They just do what the educator want, like asked to the library. They go to generally there but nothing reading seriously. Any students feel that reading through is not important, boring as well as can't see colorful pics on there. Yeah, it is for being complicated. Book is very important for yourself. As we know that on this age, many ways to get whatever we want. Likewise word says, many ways to reach Chinese's country. Therefore this Clean Code: A Handbook of Agile Software Craftsmanship can make you feel more interested to read.

Angeline Allison:

What is your hobby? Have you heard which question when you got scholars? We believe that that question was given by teacher for their students. Many kinds of hobby, Everybody has different hobby. And also you know that little person just like reading or as reading through become their hobby. You have to know that reading is very important and book as to be the matter. Book is important thing to include you knowledge, except your current teacher or lecturer. You will find good news or update in relation to something by book. Numerous books that can you choose to use be your object. One of them is actually Clean Code: A Handbook of Agile Software Craftsmanship.

[GET]? Clean Code: A Handbook of Agile Software Craftsmanship Robert C. Martin

NWIIYBFR4Z8K